

What is claimed is:

1. A software processing method comprising:

a monitoring step for status of use, which monitors the status of use of a resource to be used by a processor, and
5 an altering step for software processes, which appropriately changes a software processing method to be executed in response to contention information obtained by the monitoring step for the status of use.

10 2. The software processing method according to claim 1, wherein: the altering step for software processes comprises a plurality of executing methods that allow software to execute a process, and one of the executing methods is selected when the software is executing the process in response to contention
15 information obtained by the monitoring step for the status of use.

3. The software processing method according to claim 1, wherein: the resource is a storing device for a process, and
20 the monitoring step for the status of use monitors the status of use of the storing device.

4. The software processing method according to claim 2, wherein: the resource is a storing device for a process, and
25 the monitoring step for the status of use monitors the status of use of the storing device.

5. The software processing method according to claim 3, wherein: the monitoring step for status of use stores the preceding statuses of use of the storing device corresponding
30 to a plurality of previous clocks so that the contention information is generated based upon the previous and current statuses of use.

6. The software processing method according to claim 4,
wherein: the monitoring step for status of use stores the
preceding statuses of use of the storing device corresponding
5 to a plurality of previous clocks so that the contention
information is generated based upon the previous and current
statuses of use.

7. The software processing method according to claim 3,
10 wherein: the monitoring step for status of use stores the time
of use when the storing device is in use, and based upon whether
or not the time of use is not less than a predetermined value,
the contention information is generated.

15 8. The software processing method according to claim 4,
wherein: the monitoring step for status of use stores the time
of use when the storing device is in use, and based upon whether
or not the time of use is not less than a predetermined value,
the contention information is generated.

20 9. The software processing method according to claim 1,
wherein: the resource comprises a storing device for a process
and a bus that connects the processor to the storing device,
and the monitoring step for status of use monitors the status
25 of use of the bus.

10. The software processing method according to claim 2,
wherein: the resource comprises a storing device for a process
and a bus that connects the processor to the storing device,
30 and the monitoring step for status of use monitors the status
of use of the bus.

11. The software processing method according to claim 9,

wherein: the monitoring step for status of use stores the preceding statuses of use of the bus corresponding to a plurality of previous clocks so that the contention information is generated based upon the previous and current statuses of use.

12. The software processing method according to claim 10, wherein: the monitoring step for status of use stores the preceding statuses of use of the bus corresponding to a plurality of previous clocks so that the contention information is generated based upon the previous and current statuses of use.

13. The software processing method according to claim 9, wherein: the monitoring step for status of use stores the time of use when the bus is in use, and based upon whether or not the time of use is not less than a predetermined value, the contention information is generated.

14. The software processing method according to claim 10, wherein: the monitoring step for status of use stores the time of use when the bus is in use, and based upon whether or not the time of use is not less than a predetermined value, the contention information is generated.

15. The software processing method according to claim 1, wherein: the resource is a second processor that executes a process in response to a processing request from the processor, and the monitoring step for status of use monitors the status of use of the second processor.

16. The software processing method according to claim 15, further comprising a plurality of memory banks that are accessed

by using the same address,

wherein contention information that is obtained from the monitoring step for status of use is a signal that indicates selection of one of the memory banks.

5

17. The software processing method according to claim 1, further comprising: a compiler

wherein the compiler adds to software the following means and processes: a process identifying means which identifies
10 whether or not a process uses the resource from software; an equivalent process that is equivalent to the process identified by the process identifying means, and does not use the resource; a determining process for the status of use, which determines the status of use based upon contention information obtained
15 in the step of monitoring the status of use; and a substituting process which substitutes the equivalent process appropriately for the process identified by the process-identifying means based upon the results of the determining process for the status of use.

20

18. The software processing method according to claim 1, further comprising: a compiler

wherein the compiler adds to software the following means and processes: a process identifying means which identifies
25 whether or not a process uses the resource from the software; an equivalent process that is equivalent to the process identified by the process identifying means, and does not use the resource; a storing process for storing contention information obtained in the step of the monitoring process for
30 status of use at the current time; a determining process for determining the status of use based upon the contention information at the past time; and a substituting process which substitutes the equivalent process appropriately for the

process identified by the process-identifying means based upon the results of the determining process for the status of use.

19. The software processing method according to claim 17,
5 wherein: the contention information is processing time from the issuance of the processing request for the resource until the completion of the process, and the determining process for the status of use is a process which compares the processing time to a preset value.

10 20. The software processing method according to claim 18, wherein: the contention information is processing time from the issuance of the processing request for the resource until the completion of the process, and the determining process for the
15 status of use is a process which compares the processing time to a preset value.

21. The software processing method according to claim 17,
20 wherein: the contention information is waiting time from the issuance of the processing request for the resource until the start of the process, and the determining process for the status of use is a process which compares the waiting time to a preset value.

25 22. The software processing method according to claim 18, wherein: the contention information is waiting time from the issuance of the processing request for the resource until the start of the process, and the determining process for the status of use is a process which compares the waiting time to a preset
30 value.

23. The software processing method according to claim 17, wherein the determining process for status of use reexamines

the determination for the status of use of the resource regularly or irregularly.

24. The software processing method according to claim 18,
5 wherein the determining process for status of use reexamines the determination for the status of use of the resource regularly or irregularly.

25. The software processing method according to claim 23,
10 wherein the determining process for status of use reexamines the determination for the status of use of the resource by using random numbers.

26. The software processing method according to claim 24,
15 wherein the determining process for status of use reexamines the determination for the status of use of the resource by using random numbers.

27. The software processing method according to claim 18,
20 wherein: in the case when processes to be extracted by the process-identifying means are extracted from a plurality of portions of the software, the compiler further adds to the software an identifying process for identifying the portions of appearance of the processes identified by the process-
25 identifying means, and the storing process stores the contention information for each of the portions of appearance so that the determining process for status of use carries out the determination by using the contention information stored for each of the portions of appearance.

30

28. A software processing system comprising:
a processor;
a resource that the processor uses;

a monitoring device for status of use which monitors status of use of the resource;

an altering device for a software process which appropriately alters a processing method of software to be
5 executed in response to contention information obtained by the monitoring device for the status of use.

29. The software processing system according to claim 28,
wherein: the altering device for a software process prepares
10 a plurality of executing methods to be used by the software to execute a process, and selects one of the executing methods in response to the contention information when the software is in operation.

15 30. The software processing system according to claim 28, wherein: the resource is a second processor that carries out a process in accordance with a processing request from the processor, and the monitoring device for status of use monitors the status of use of the second processor.

20 31. The software processing system according to claim 30, wherein the contention information is an interrupt signal to the processor.